An Algorithm to Select Target Specific Probes for DNA Chips

Lars Kaderali*
kaderali@zpr.uni-koeln.de

Alexander Schliep
schliep@zpr.uni-koeln.de

Center for Applied Computer Sciences Cologne (ZAIK)
University of Cologne
Weyertal 80, 50931 Köln, Germany

Keywords: Primer Design, DNA Arrays, Probe Selection

Abstract

Motivation: The selection of target specific probes is a relevant problem in the design of DNA chips. Given a set S of genomic sequences, the task is to find at least one oligonucleotide, called probe, for each target sequence in S. This probe will be attached to the chip surface, and must be chosen in a way that it will not hybridize to any other sequence but the intended target. Furthermore, all probes on the chip must hybridize to their intended targets under the same reaction conditions, most importantly at the temperature T at which the experiment is conducted.

Results: We present an efficient algorithm for the probe design problem. Melting temperatures are calculated for all possible probe-target interactions using an extended nearest-neighbor model, allowing for both non-Watson-Crick base-pairing and unpaired bases within a duplex. To compute temperatures efficiently, a combination of suffix trees and dynamic programming based alignment algorithms is introduced. Additional filtering steps during preprocessing increase the speed of the computation. Also, an algorithm to select the actual probes from the set of candidates is presented.

The practicability of the algorithms is demonstrated by two case studies: The computation of probes for the identification of different HIV-1 subtypes, and finding probes for 28S rDNA sequences from over 400 organisms.

Availability: The software is available to academic users on request from the authors.

Contact: {kaderali,schliep}@zpr.uni-koeln.de

Supplementary Information: http://www.zaik.uni-koeln.de/bioinformatik/arraydesign.html

*To whom correspondence should be addressed

Introduction

Both in medicine and biology efficient diagnostic tests to probe genetic information and measure tissue- or cell-specific expression of hereditary information are required. The availability of sequences of complete genomes permits interesting questions to be asked and answered at the genome level rather than at the level of the individual gene. Unfortunately, traditional tools such as the polymerase chain reaction (PCR) and diverse blotting techniques do not scale well enough to efficiently support the size of assays required for such tasks. For this reason, DNA chips have gained wide use in biological research.

For DNA chip experiments to succeed, appropriate probes have to be selected for each individual spot on the chip surface. Given a set of genomic sequences, the target sequences, we have to find at least one probe for every target sequence in the set. These probes will then be attached to the chip surface. Each probe on the chip should hybridize only to the intended target, and not to any other sequence in the target set, i.e., a probe must have a high specificity in detecting the target. The problem is further complicated as all probes must work under the same hybridization conditions, most importantly, at the same temperature. The problem can be formalized as follows:

Given n target sequences $t_1, t_2, ..., t_n$, find a temperature T and n probe sequences $p_1, p_2, ..., p_n$ such that

$$T_M(p_i, t_i) - \epsilon > T > T_M(p_i, t_k) + \epsilon \qquad (1)$$

for all $k \neq i, i = 1, ..., n$, where $T_M(x, y)$ is the temperature below which the two strands x and y are bound, and above which they denature. T is the temperature at which the chip experiment should be carried out. The additional temperature margin ϵ compensates for example for model errors and imprecisions.

Melting Theory and Nearest Neighbor Model

The computation of T_M for a given duplex is based on the assumption that we deal with two-state transitions: Either the DNA is in the double helical state, or it is in the random coil, denatured state. We consider the two-state reversible equilibrium annealing reaction of two DNA single strands (compare (Owczarzy *et al.*, 1997))

$$S_1 + S_2 \overset{K_D}{\rightleftharpoons} D. \tag{2}$$

where K_D is the equilibrium constant.

T_M is defined as the temperature at which 50% of the strands are in the double stranded and 50% in the random coil, denatured state. It can be shown that (compare (Freier *et al.*, 1986; Kaderali, 2001; Ornstein & Fresco, 1983; Owczarzy *et al.*, 1997; Rychlik & Rhoads, 1989))

$$T_M = \frac{\Delta H}{\Delta S + R \ln C_T/4}, \tag{3}$$

where ΔH and ΔS are enthalpy and entropy changes of the nucleation reaction, R is the Boltzmann constant, and $C_T = [S_1] + [S_2] + 2[D]$ is the total molar concentration of strands.

This concentration dependence of T_M induces some problems to our ansatz of calculation, as target DNA concentration is unknown in DNA array experiments. Thus the calculation cannot be accurate. However, (Li & Stormo, 2000) report that T_M is still sufficiently precise for probe evaluation. They suggest using a constant of $1 \times 10^{-6} M$ for C_T.

Interactions between bases in nucleic acids are of two kinds (Cupal, 1997):

- *Base pairing* in the plane of the bases due to hydrogen bonding between base pairs in the two opposing strands, and
- *Base stacking* perpendicular to the plane of the bases due to London dispersion forces and hydrophobic effects.

Both quantum chemical calculations and thermodynamic measurements suggest that base pairing contributions to total energy depend exclusively on base pair composition, while stacking contributions depend on base pair composition and base sequence along the chain. Obviously, models based solely on base composition neglect stacking contributions, and yield less precise results (Rychlik & Rhoads, 1989).

As the major contribution to the overall stabilizing energy of nucleic acid structures results from short-range interactions, we assume that the stability of a base pair (and its contribution to enthalpy and entropy of the duplex) depends only on the identity of its immediate up- and downstream neighbors (Cupal, 1997). This assumption leads to the Nearest Neighbor (NN) Model. In this model we assume that ΔH and ΔS of the melting reaction can be calculated by summing up the contributions of the individual neighboring pairs as follows:

$$\Delta H = \sum_{i=1}^{n-1} \Delta H_{x_i, x_{i+1}/y_i, y_{i+1}}, \text{ and} \tag{4}$$

$$\Delta S = \sum_{i=1}^{n-1} \Delta S_{x_i, x_{i+1}/y_i, y_{i+1}}, \tag{5}$$

where x_i denotes the i-th base in strand one read in 5'-3' direction, whereas y_j refers to the j-th base of strand two read in 3'-5' direction. ΔH and ΔS can then be used with equation (3) to calculate the melting temperature of the strands.

Usually, thermodynamic parameters for the nearest neighbor model are determined from UV-absorbance-vs-temperature profiles of a number of different, short oligonucleotides. By fitting the measured curves to the model, parameters can be obtained that according to SantaLucia on average fit ΔG, ΔH, ΔS, and T_M within 4%, 7%, 8% and 2 degrees Celsius, respectively (SantaLucia Jr. *et al.*, 1996). Parameters are available for DNA-DNA (Allawi & SantaLucia, 1997; Allawi & SantaLucia, 1998*a*; Allawi & SantaLucia, 1998*b*; Allawi & SantaLucia, 1998*c*; Breslauer *et al.*, 1986; Gotoh & Tagashira, 1981; Peyret *et al.*, 1999; Quartin & Wetmur, 1989; SantaLucia Jr. *et al.*, 1996; SantaLucia, 1998; Sugimoto *et al.*, 1996), RNA-RNA (Freier *et al.*, 1986; SantaLucia Jr. & Turner, 1997; Xia *et al.*, 1998) and DNA-RNA (Gray, 1997) duplexes; by using the appropriate set of parameters, the nearest neighbor model can be applied to all these cases.

Algorithm

To select optimum probes, melting temperatures between the complements of all substrings of all target sequences (as the probe candidates) and all targets have to be computed. To apply the NN model, an alignment of the probe and the target sequence is required, and the alignment resulting in the highest T_M is desired. Furthermore, as it is possible that the probe–target-duplex contains secondary structure, all combinations containing loops et cetera should be considered as well.

By excluding bad probe candidates as early as possible, running time can be reduced considerably, as then no alignments have to be computed for that candidate. Note also, that prefixes shared between different substrings are

quite common in DNA. Hence, additional running time can be saved by avoiding to recompute entropy and enthalpy values for duplexes involving such prefixes.

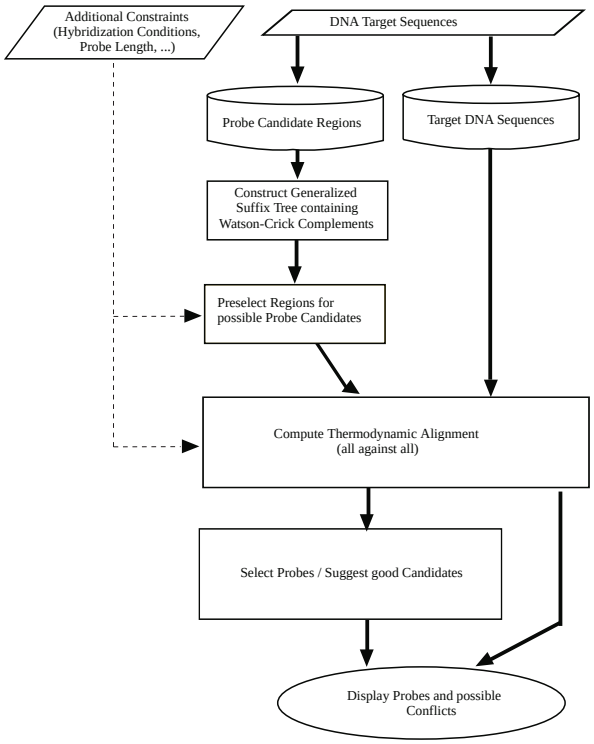

Figure 1: Method overview for probe selection algorithm

Figure 1 gives an overview of the probe design algorithm. Given the DNA target sequences (we speak of DNA here and in the following, although the algorithm works just as well for RNA, if the appropriate parameter sets are used), our goal is to exclude infeasible probe candidates as early as possible. Infeasible probes are probes that are too short or too long, occur more than once in different probes, or do not fulfill other relevant criteria. Thermodynamic conputations will only be done for the remaining probes.

The algorithm begins by constructing a generalized suffix tree from the inverse complements of all target sequences. A suffix tree is a data structure allowing for fast recognition of repetitive subsequences in strings. This property is used to identify non-unique probes, i.e., probes forming perfect duplexes with more than one target, which are subsequently removed from the probe candidate set. Also, some other criteria are used to remove infeasible probes, such as the probe length and the melting temperature of the probe with its perfect Watson-Crick complement. We will come back to this point later. Note, that additional criteria can be included easily, and that it is possible to remove entire regions based on, for example, some biological background knowledge on the sequences under consideration.

Given the target DNA sequences and probe candidates, the algorithm computes melting temperatures for all combinations of probe–target interactions, i.e., melting temperatures between all probe candidates paired with all target sequences. As DNA is known to be highly repetitive (compare (Gusfield, 1997), page 286), much time can be saved by avoiding recomputation of melting temperatures for subdomains of probes with some given target that have already been considered. The probes are stored in a generalized suffix tree in the preselection step, and this suffix tree is used further in the algorithm to avoid such redundant computations.

Finally, probes and melting temperatures are output, and chip probes can be picked from suggestions made or evaluated further by sorting output data and, for example, graphically visualizing crosshybridization conflicts.

Thermodynamic Alignment

To apply the nearest neighbor model, we need to know which bases are going to form basepairs in the duplex. Unfortunately, this is not clear at all if the strands are not perfectly complementary to one another in the Watson-Crick sense. Worse yet, bases may remain unpaired within a duplex, and the duplex will still be quite stable (Ke & Wartell, 1995; LeBlanc & Morden, 1991; Turner, 1992). Hence, an alignment of the two sequences is required, where gaps should be allowed. The alignment and T_M are interdependent: We cannot compute T_M without knowing the alignment, and the alignment should maximize T_M. Enumerating all possible alignments and computing their respective melting temperatures to choose the maximum thereof is infeasible, as the number of alignments grows exponentially with sequence length and the problem would quickly become computationally intractable.

The problem of aligning two sequences given a weight function $w(\cdot,\cdot)$ is one of the standard bioinformatics problems. A dynamic programming algorithm due to Needleman-Wunsch (Durbin *et al.*, 1989; Gusfield, 1997; Waterman, 1995) can be used to find an alignment maximizing $w(\cdot,\cdot)$. The general idea is to consecutively extend the alignment, starting with an alignment of prefixes of the two sequences x and y. This is done by creating a table A, writing x on one axis and y on the other. Element $A_{i,j}$ of the table stores the optimum score for an alignment of the prefixes $x[1..i]$ and $y[1..j]$, the prefixes consisting of the first i and j characters of x and y, respectively. As the weight of an alignment is an additive function of the individual base-pair weights, the simple

recursion

$$A_{i,j} := \max \begin{cases} A_{i-1,j-1} + w(x_i, y_j) \\ A_{i-1,j} + w(x_i, -) \\ A_{i,j-1} + w(-, x_j) \end{cases} \quad (6)$$

allows to compute all values in table A correctly (provided the "border" $A_{0,*}$ and $A_{*,0}$ has been initialized properly). The weight of the optimum alignment can be found in $A_{i,j}$.

We have modified this algorithm to calculate ΔH and ΔS at every position in the dynamic programming table, choosing the prefix-alignment resulting in the highest local melting temperature: Our cost function is the T_M function from equation (3), and instead of storing T_M in the dynamic programming table, we store values for ΔH and ΔS at every position in the table. Then, our recursion becomes:

$$\Delta H_{i,j} = \begin{cases} \Delta H_{i-1,j-1} + \Delta\Delta H(x_i, y_j) & \text{if } t = 0 \\ \Delta H_{i-1,j} + \Delta\Delta H(x_i, -) & \text{if } t = 1 \\ \Delta H_{i,j-1} + \Delta\Delta H(-, y_j) & \text{if } t = 2 \end{cases} \quad (7)$$

$$\Delta S_{i,j} = \begin{cases} \Delta S_{i-1,j-1} + \Delta\Delta S(x_i, y_j) & \text{if } t = 0 \\ \Delta S_{i-1,j} + \Delta\Delta S(x_i, -) & \text{if } t = 1 \\ \Delta S_{i,j-1} + \Delta\Delta S(-, y_j) & \text{if } t = 2 \end{cases} \quad (8)$$

and $t \in \{0, 1, 2\}$ is to be chosen such that

$$T_M = \frac{\Delta H_{i,j}}{\Delta S_{i,j} + R \ln C_T/4} \quad (9)$$

is maximal. Note, that $\Delta\Delta H(x_i, y_j)$ and $\Delta\Delta S(x_i, y_j)$ denote the values from the nearest neighbor parameters for enthalpy and entropy changes, respectively, when the i-th base of x, x_i and the j-th base of y, y_j are paired in the alignment. "–" stands for a gap in the alignment, representing an unpaired base in the duplex. Note also, that $\Delta\Delta H$ and $\Delta\Delta S$ depend not only on the current basepair, but also on the one before (the nearest neighbor). However, implementing this dependency is straightforward. We neglect this issue here for the sake of simplicity.

By initializing the border of the dynamic programming table with zeros, we assure that initial gaps do not lower T_M; by looking for the result not just in cell $(|x|, |y|)$, but in cells $(s, |y|)$ and $(|x|, t)$ for all $s = 1..|x|$ and $t = 1..|y|$ and choosing the maximum value found, the same is true for terminal gaps.

Unfortunately, as the melting temperature equation (9) is not linear, the alignment algorithm is not guaranteed to return the optimum alignment with the highest T_M possible. To assess the quality of the approximation (using the parameters listed in (Kaderali, 2001)), we have enumerated all perfect Watson-Crick duplexes of length up to 15 nucleotides, and shown that the algorithm finds the optimum alignment in all these cases. Furthermore, for over 100,000 random Watson-Crick duplexes of length up to 250, not a single error was made either. In the case where the most stable duplex contains one single unpaired nucleotide, the greedy approach may fail. Consider, for example, the duplex

```
0 1 2 3 4 5 6 7 8 9 a b
$ G T G T G C A A A A $
- $ C C A C G T T T T $
. . M M M M M M M M M M
```

with a melting temperature $T_M = 27.2°$C, whereas the alignment algorithm finds

```
0 1 2 3 4 5 6 7 8 9 a b
$ G T G T G C A A A A $
$ C - C A C G T T T T $
. M . M M M M M M M M .
```

with $T_M = 15.4°$C. The difference is caused when the algorithm is forming the G/C pair in position 3. It has to decide between either the \$GT/\$C- alignment or the \$GT/-\$C alignment. The alignment resulting in the higher local melting temperature is chosen — but unfortunately, when more bases are added after the G/C pair, it turns out that the wrong choice has been made.

To be able to estimate the magnitude of the error for more distant sequences, two experiments involving randomly generated sequences have been performed:

1. Generate two random sequences of random lengths between *minlen* and *maxlen* nucleotides; note that the two sequences generated may be of different length. Run the thermodynamic alignment algorithm to calculate the alignment melting temperature T_M^{align}. In parallel, enumerate all possible alignments, calculate their respective melting temperatures, and save the maximum T_M^{enum} thereof.

2. Generate one random sequence of length between *minlen* and *maxlen*. Then construct a second sequence as the inverse Watson-Crick complement thereof (meaning that this second sequence forms a perfect duplex with the former one), and introduce at most *maxmut* insertions, deletions or substitutions. Again, run the thermodynamic alignment algorithm to calculate the alignment melting temperature T_M^{align}. In parallel, enumerate all possible alignments, calculate their respective melting temperatures, and save the maximum T_M^{enum} thereof.

Experiment 1 has been carried out with $minlen = 10$ and $maxlen = 15$ for 2500 random sequences. The results are reassuring. For 1241 alignments or 49.64%, T_M^{align} and T_M^{enum} were equal. For another 272 alignments (= 10.88%), an error of at most three degrees Celsius was made, which is about the average error inherent

in the nearest neighbor model. For 79.76% of the alignments, the error was no more than 10 degrees Celsius. Figure 2 depicts the difference $T_M^{enum} - T_M^{align}$ (rounded **up** to the next integer) versus the number of sequences with that error. The average error made was 3.13 degrees Celsius, the maximum error observed was 35.46 degrees.

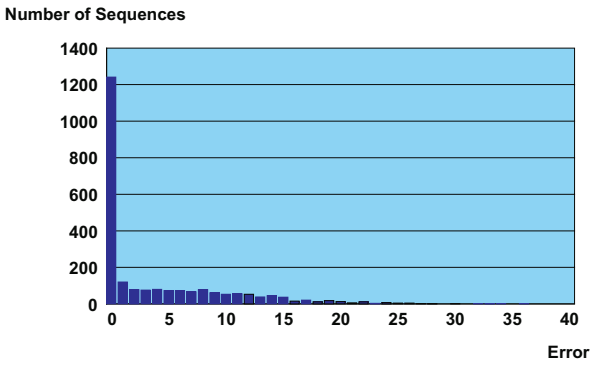

Figure 2: The diagram shows the error made (*rounded up* to the next integer) vs the number of alignments with an error of that magnitude, for 2500 alignments of random sequences of lengths between 10 to 15 nucleotides.

The case where the calculated temperature is too low when the sequences forming the duplex have only little similarity is not really a problem — the actual melting temperature of the duplex will be too low to play a role in probe design anyways. Therefore, the results of experiment 2 are even more interesting. Again, the experiment was conducted for 2500 alignments with at most one mutation. In that case, no error was made in 87.12% of the cases. An error of not more than three degrees was made in 90.12% of the alignments, and an error of at most 10 degrees was made in 93.84% of the alignments. The average error made was 1.27°C, the maximum error was 34.3 degrees Celsius.

Using dynamic programming, the thermodynamic alignment algorithm realizes a considerable reduction of running time by avoiding enumeration of all possible alignments for two given sequences. However, under certain circumstances we lose optimality.

Suffix Trees

The algorithmic idea introduced in this section will reduce running time as well. Nothing is lost in terms of result quality, all we need is a little more memory and an additional data structure.

The underlying idea is straightforward. Assume we have just computed the dynamic programming table for the two sequences "GATTACA" and "CTAAGGT". Further assume we need to align "GATTACA" and "CTAATGA" sometime thereafter. Then, the two dynamic programming tables share the subtable for "GATTACA" and "CTAA". We need not recompute this part of the dynamic programming table for the latter alignment, but may use the subtable from the former alignment and compute only the remaining, different entries. To identify common prefixes of substring pairs of *all* the different sequences under consideration, we use a generalized suffix tree. This tree is then used in both probe preselection and alignment.

A suffix tree for the sequence "TACTACA" is shown in Figure 3. Note that, by appending the unique character "$" at the end of the string, we guarantee that every suffix ends at a leaf. Otherwise, the suffix "A" of "TACTACA", i.e., the suffix consisting of only the last character of the original sequence, would end within the "AC" edge. This problem arises whenever a suffix of the string is a prefix of another suffix.

Although suffix trees appear to be quite complex at first sight, they can surprisingly be constructed in linear time in the length of the given string. Esko Ukkonen (Ukkonen, 1995) devised a straightforward $\mathcal{O}(n)$ algorithm in 1995. An excellent description of that algorithm can be found in (Gusfield, 1997).

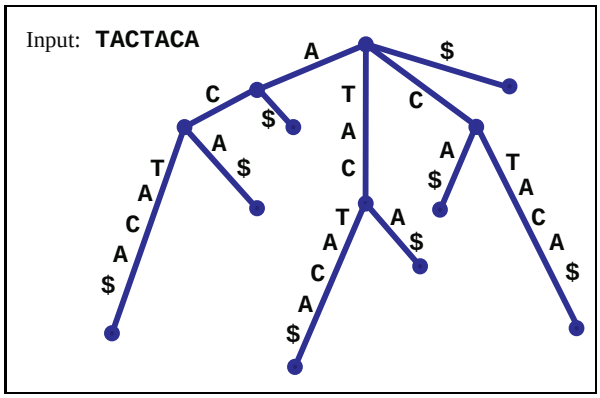

Figure 3: Suffix tree for the sequence "TACTACA". Note how all the suffixes "TACTACA", "ACTACA", "CTACA", "TACA","ACA","CA", "A" and the empty suffix are described by a unique path from the root node to one of the leaves, and how every leaf uniquely yields one such suffix. The symbol "$" denotes the end of a string.

Suffix trees represent all suffixes of one given string. A *Generalized Suffix Tree* is a suffix tree containing all suffixes of a finite number of strings. Only slight modifications are required to construct generalized suffix trees with Ukkonen's algorithm, and the resulting algorithm still runs in linear time.

Probe Preselection

As mentioned before, we need to compute melting temperatures (and alignments) between all probe candidates and all sequences in the target set. Therefore, it seems worthwhile to put some effort into reducing the number of probe candidates as early as possible. There are several criteria that help exclude infeasible probes:

- *Probe Length:* Usually, there are some restrictions to probe length. These may be due to technical limitations in the process of chip manufacturing, as well as limitations given by the user or other external causes. Without going into more detail at this point, we assume that we have variables $minlen$ and $maxlen$ with $minlen \leq |probe_i| \leq maxlen$, where $|probe_i|$ is the length of probe i, and all feasible probes have to satisfy that inequality. Clearly, it makes no sense to align a probe p to any target sequence if p is too long. Similarly, if $|p| < minlen$, the probe can be skipped as well.

- *Unique Probes:* If a given probe is the perfect Watson-Crick complement to substrings of two or more target sequences, that probe will hybridize to both targets with the same melting temperature. Therefore, such probes cannot be used for chip experiments, as both targets would hybridize against the same spot on the chip. We will allow only probes that are complementary to exactly one substring of all target sequences.

- *Probe Melting Temperature:* Last but not least, one can impose some constraints on the minimum temperature that a probe–target duplex should be able to withstand. The chip experiment will be carried out at some temperature T, therefore $T_M(target, probe) > T$ must hold. Of course, the problem of determining T and the probes to be used are not independent from one another. However, we assume some bound $T_B \leq T$ to be given that can be used to exclude probes with $T_M(probe, target) < T_B$ from further consideration.

The algorithm to preselect probes starts with the set of complements of all substrings of all the target sequences. Every substring of a string is a prefix of a suffix of that string. Therefore, a generalized suffix tree can be used to represent all substrings. By following a unique path from the root to another node, either leaf or internal, all substrings can be retrieved from the tree (the path may end somewhere within an edge, i.e., it need not necessarily terminate at a node). Similarly, each substring corresponds to one such path starting at the root node. Note, however, that one path may correspond to two (equal) substrings from the same or different sequences.

After the generalized suffix tree containing the complements of all target sequences has been constructed, applying the above criteria and removing all infeasible probes from the tree is straightforward. During preprocessing, we insert additional nodes in the tree to assure that every probe candidate corresponds to a leaf. When preprocessing is done, the suffix tree has been converted to a keyword tree (Gusfield, 1997) containing all the feasible probe candidates, where each candidate corresponds to the path-label of a leaf node.

Thermodynamic Tree Alignment

Recall, that our objective is to determine a probe for each of the n genomic sequences that will hybridize only to its respective target, and not to any of the other sequences. Hence, we need to compute the melting temperatures of the most stable duplex formed between each probe left after preprocessing and each target. The final step of selecting probes from the output of the thermodynamic tree alignment algorithm will be described in the next section.

To compute the melting temperatures, begin with the complements of all substrings of the n target sequences. This is done by constructing a generalized suffix tree containing the complements of all target sequences. Then, all substrings are contained in that tree. The second step is to reduce the number of substrings stored in the tree, and to assure that all probe candidates remaining in the tree correspond to a leaf. These steps are taken care of as described in the previous section on probe preselection.

Finally, all that remains to be done is the computation of the melting temperatures of the duplexes formed between all substrings left in the tree and all target sequences; i.e., each substring and each target have to be aligned using the thermodynamic alignment algorithm described above, and the maximum melting temperature must be determined. Doing so is extremely time-consuming. We will therefore use the modified suffix tree from the preprocessing step to reduce running time.

Repetitive subsequences in DNA are quite common. Thus, whenever calculating alignments of two strings, we may be able to reuse parts of the dynamic programming table from a previously computed alignment, if the strings from that prior alignment share prefixes with the actual strings.

Fortunately, we can use the tree constructed during preprocessing to identify such common prefixes of probes. The tree induces an ordering of the probes, grouping probes with common prefixes together. This helps to calculate such groups at a time and to avoid the storage of different subtables, which reduces the overall memory requirements of the program.

Special care must be taken to initialize the tables up-

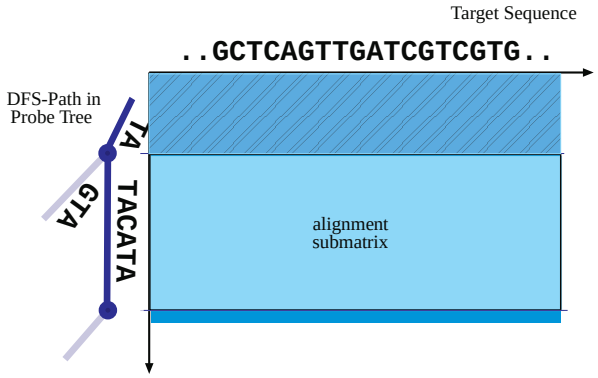

Figure 4: Alignment of the target sequence with probe $..TATACATA...$ Note that if the alignment of the sequence TAGTA and the same target has been computed before, the upper part of the dynamic programming table can be reused.

per and left border properly when a new target sequence is introduced. Initial gaps in a sequence should not be penalized. Hence the upper and left border of the dynamic programming table should be set to zero. Similarly, terminal gaps should not be penalized either. Hence, when we retrieve the maximum melting temperature from the alignment table, we look up all entries in the last row and last column of the dynamic programming table, and choose the maximum thereof.

Probe Picking

The thermodynamic tree alignment algorithm determines probe candidates according to certain criteria, and returns melting temperatures $T_M(probe, target)$ for all $(probe\ candidate, target)$ pairs, i.e., all probe candidates and all target sequences.

Given the output list from the thermodynamic tree alignment algorithm, our objective now is to select a temperature T and one probe from the list for each of the target sequences, such that the chip experiment can be carried out at temperature T, and the probes selected will hybridize only to their intended target sequence, and not to any of the other sequences. This problem can be formalized as follows:

PSP: Given n DNA or RNA target sequences $t_1, t_2, .., t_n$, given furthermore for each target sequence t_i a finite set of probe sequences $\mathcal{P}_i$, where $\mathcal{P}_i \bigcap \mathcal{P}_j = \emptyset$ for all $i, j; i \neq j$. Furthermore given for all target sequences t_i and all probe candidates $p_j \in \bigcup_{k=1}^{n} \mathcal{P}_k$ the melting temperatures $T_M(t_i, p_j)$ at which target t_i and probe p_j dissociate.

Find a temperature T and, for each target sequence t_i, select one probe $p_k \in \mathcal{P}_i$ s.t.

$$T_M(t_i, p_k) \geq T > T_M(t_j, p_k) \tag{10}$$

for all $j \neq i$.

The temperature T is a temperature that must hold for all probes selected; the inequality above must be satisfied by all probes selected for all targets with the same temperature T. This implies that for two selected probes p_i for target t_i and p_j for target t_j, the inequalities $T_M(t_i, p_i) > T_M(t_j, p_i)$, $T_M(t_i, p_i) > T_M(t_i, p_j)$, $T_M(t_j, p_j) > T_M(t_i, p_j)$ and $T_M(t_j, p_j) > T_M(t_j, p_i)$ must hold: All "desired" hybridizations have melting temperatures higher than all "undesired" cross-hybridizations.

This problem can be solved in polynomial time. The idea is to sort the probes for each target according to their melting temperature. Then, starting with the highest temperature T, consecutively lower T, and remove all probes that will crosshybridize at the new temperature. This is iterated until either a feasible, unambiguous probe is found for every probe, or until all probes have been removed. Figure 5 illustrates the procedure.

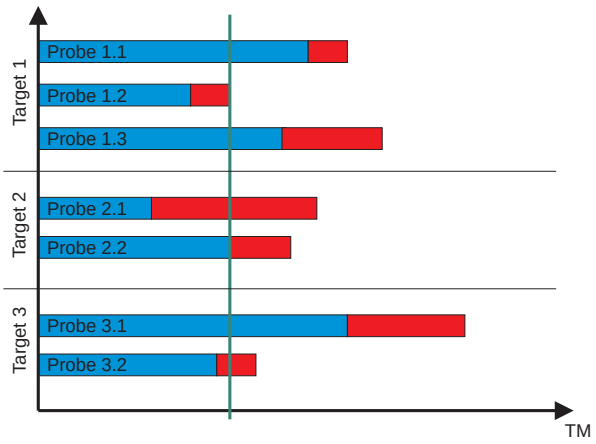

Figure 5: Probe picking. The X-axis represents the melting temperature, the Y-axis the different probes. Probe indices are of the form $t.p$, where t refers to the target sequence the respective probe is intended for, and p is the number of the probe for that target. For each probe, the right end of the dark bar shows T_M of the perfect duplex formed between the probe and its respective target, the light gray bar shows the temperature range where the probe will crosshybridize. For the temperature represented by the line vertically crossing all probes, probes 1.2, 2.1 and 3.2 would yield a feasible set of probes. The algorithm starts with the maximum melting temperature found for some duplex, and decreases T until such a set is found.

Implementation

The algorithms presented here have been implemented in C++. The *Probesel* program combines probe preselection and the thermodynamic alignment algorithm and calculates melting temperatures between probe candidates and all target sequences. The *Pickprb* program implements the Probe Selection Problem (PSP) Algorithm to select one probe for each target sequence from the output generated by *Probesel*.

The program code has been tested on Intel-PCs under Windows NT 4.0 with Microsoft's Visual C++ 6.0, on Sun Ultra Enterprise 4000 running Solaris 7 with the GNU g++ compiler, and on DEC Alpha / Compaq Tru64 UNIX V5.1 with Compaq's cxx compiler, version 6.20.

Discussion

Identifying HIV-1 Subtypes

Besides running on randomly generated sequences of different lengths, the algorithm has been used to identify optimum probes to be used for the identification of different HIV-1 subtypes. The complete HIV-1 reference subtypes database from Los Alamos National Laboratories, USA (LANL, 1999) has been processed. This database contains 58 sequences of average length around 9300 nucleotides. All probes of length 20 with a melting temperature above 70°C have been evaluated. The entire computation took 8.7 hours for the *probesel* program and a couple of seconds for *pickprb* on a Compaq Tru64 machine with four DEC Alpha EV6.7 processors each operating at 667 MHz, and equipped with an alpha internal floating point processor. The machine has a total of 6017M RAM. Note that the present version of the program runs single-threaded and hence makes no use of the multiple processors available.

All possible duplexes with a temperature above 0°C have been written to a file by *probesel*, which was then processed by *pickprb*. Probes were found for all 58 sequences, with melting temperatures between 73.4° and 84.8° Celsius. The highest temperature for which crosshybridizations are predicted is 53.3° Celsius, which gives a margin of more than twenty degrees. The program suggests conducting the experiment at a temperature of 63.5 degrees Celsius.

Application to 28S rDNA Sequences

The algorithms presented here have been applied to a database of 1230 28S rDNA sequences from different organisms (Markmann, 2000). Those 1230 sequences are of length between 160 and 6198 bases, with an average length of 676 nucleotides. As the database contains sequences with very high similarity ($> 95\%$), it was filtered before starting the *Probesel* program. To do so, pairwise alignments of all sequences were computed, using edit distance as distance function. Then, for each aligned pair of sequences, all matches between the two sequences were counted. This was set in relation to the length of each of the sequences in the alignment, including internal gaps, but not counting initial and terminal gaps. Whenever some sequence was over 95% similar to another sequence according to that metric, it was removed. If both sequences had relative similarity of over 95% to one another, the shorter one was removed.

487 sequences remained in the database after this preprocessing step. Then, the probesel algorithm was started with probe length 29-30 and minimum probe-target melting temperature 60°C. No unique probes could be found for 44 Sequences, which *Probesel* reported after approximately 2 minutes. These sequences were removed from the target set, and the program was started again.

The calculation of all melting temperatures — including the preprocessing step— took 60.6 hours, or two and a half days. Then, probe picking was completed in less than a minute. The algorithm suggests conducting the chip experiment at a temperature of 70.3 degrees Celsius, and finds probes for 396 of the 443 targets, when a margin of five degrees is enforced between the highest crosshybridization temperature and the lowest temperature at which intended hybridizations occur. The lowest "intended hybridization" T_M was 72.8 degrees, whereas the highest temperature calculated for crosshybridizations was 67.7 degrees celsius.

Experimental evaluation of the probes selected by the algorithm is under way. Results will be published elsewhere.

Acknowledgements

We would like to thank Diethard Tautz, Institute of Genetics, University of Cologne for pointing out the problem and helping with various questions concerning biology, biochemistry and genetics. Alexander Pozhitkov contributed greatly by his readiness to spend hours discussing hybridization properties of nucleic acids, and Melanie Markmann provided her 28S rDNA database to test the algorithm.

References

Allawi, H. & SantaLucia, J. (1997) Thermodynamics and nmr of internal g-t mismatches in dna. *Biochemistry,* **36**, 10581–10594.

Allawi, H. & SantaLucia, J. (1998*a*) Nearest neighbor parameters for internal g-a mismatches in dna. *Biochemistry,* **37**, 2170–2179.

Allawi, H. & SantaLucia, J. (1998*b*) Nearest neighbor parameters of internal a-c mismatches in dna: sequence dependence and ph effects. *Biochemistry,* **37**, 9435–9444.

Allawi, H. & SantaLucia, J. (1998*c*) Thermodynamics and nmr of internal c-t mismatches in dna. *Nucleic Acids Research,* **26**, 2694–2701.

Breslauer, K., Frank, R., Blöcker, H. & Marky, L. (1986) Predicting dna duplex stability from the base sequence. *Proc. Natl. Acad. Sci. USA,* **83**, 3746–3750.

Cupal, J. (1997). The density of states of rna secondary structures. Master's thesis University of Vienna.

Durbin, R., Eddy, S., Krogh, A. & Mitchison, G. (1989) *Biological Sequence Analysis.* Cambridge University Press, Cambridge.

Freier, S., Kierzek, R., Jaeger, J., Sugimoto, N., Caruthers, M., Neilson, T. & Turner, D. (1986) Improved free-energy parameters for predictions of rna duplex stability. *Proc. Natl. Acad. Sci. USA,* **83**, 9373–9377.

Gotoh, O. & Tagashira, Y. (1981) Stabilities of nearest-neighbor doublets in double-helical dna determined by fitting calculated melting profiles to observed profiles. *Biopolymers,* **20**, 1033–1042.

Gray, D. (1997) Derivation of nearest-neighbor properties from data on nucleic acid oligomers. ii. thermodynamic parameters of dna-rna hybrids and dna duplexes. *Biopolymers,* **42**, 795–810.

Gusfield, D. (1997) *Algorithms on Strings, Trees, and Sequences. Computer Science and Computational Biology.* Cambridge University Press, Cambridge.

Kaderali, L. (2001). Selecting target specific probes for dna arrays. Master's thesis University of Cologne.

Ke, S. & Wartell, R. (1995) Influence of neighboring base pairs on the stability of single base bulges and base pairs in a dna fragment. *Biochemistry,* **34**, 4593–4600.

LANL (1999). Hiv sequence database - 1999 hiv-1 subtype reference alignments. Los Alamos National Laboratories, http://hiv-web.lanl.gov/.

LeBlanc, D. & Morden, K. (1991) Thermodynamic characterization of deoxyribooligonucleotide duplexes containing bulges. *Biochemistry,* **30**, 4042–4047.

Li, F. & Stormo, G. (2000) Selecting optimum dna oligos for microarrays. In *IEEE International Symposium on Bio-Informatics and Biomedical Engineering (BIBE)*, Key Bridge Marriott, Arlington, USA.

Markmann, M. (2000). *Entwicklung und Anwendung einer 28S rDNA-Sequenzdatenbank zur Aufschlüsselung der Artenvielfalt limnischer Meiobenthosfauna im Hinblick auf den Einsatz moderner Chiptechnologie.* PhD thesis, University of Munich.

Ornstein, R. & Fresco, J. (1983) Correlation of t_m and sequence of dna duplexes with δh computed by an improved empirical potential method. *Biopolymers,* **22**, 1979–2000.

Owczarzy, R., Vallone, P., Gallo, F., Paner, T., Lane, M. & Benight, A. (1997) Predicting sequence-dependent melting stability of short duplex dna oligomers. *Biopolymers,* **44**, 217–239.

Peyret, N., Seneviratne, P., Allawi, H. & SantaLucia, J. (1999) Nearest-neighbor thermodynamics and nmr of dna sequences with internal a-a, c-c, g-g and t-t mismatches. *Biochemistry,* **38**, 3468–3477.

Quartin, R. & Wetmur, J. (1989) Effect of ionic strength on the hybridization of oligonucleotides with reduced charge due to methylphosponate linkages to unmodified oligodeoxynucleotides containing the complementary sequence. *Biochemistry,* **28**, 1040–1047.

Rychlik, W. & Rhoads, R. (1989) A computer program for choosing optimal oligonucleotides for filter hybridization, sequencing and in vitro amplification of dna. *Nucleic Acids Research,* **17**, 8543–8551.

SantaLucia, J. (1998) A unified view of polymer, dumbbell, and oligonucleotide dna nearest-neighbor thermodynamics. *Proc. Natl. Acad. Sci. USA,* **95**, 1460–1465.

SantaLucia Jr., J., Allawi, H. & Seneviratne, P. (1996) Improved nearest-neighbor parameters for predicting dna duplex stability. *Biochemistry,* **35**, 3555–3562.

SantaLucia Jr., J. & Turner, D. (1997) Measuring the thermodynamics of rna secondary structure formation. *Biopoly,* **44**, 309–319.

Sugimoto, N., Nakano, S., Yoneyama, M. & Honda, K. (1996) Improved thermodynamic parameters and helix initiation factor to predict stability of dna duplexes. *Nucleic Acids Research,* **24**, 4501–4505.

Turner, D. (1992) Bulges in nucleic acids. *Current Opinion in Structural Biology,* **2**, 334–337.

Ukkonen, E. (1995) On-line construction of suffix-trees. *Algorithmica,* **14**, 249–260.

Waterman, M. (1995) *Introduction to Computational Biology.* Cambridge University Press, Cambridge.

Xia, T., SantaLucia, J., Burkard, M., Kierzyk, R., Schroeder, S., Jiao, X., Cox, C. & Turner, D. (1998) Thermodynamic parameters for an expanded nearest-neighbor model for formation of rna duplexes with watson-crick base pairs. *Biochemistry,* **37**, 14719–14735.